METHOD AND APPARATUS FOR GENERATING REPORTS USING DECLARATIVE TOOLS

This is a continuation of application Ser. No. 08/282,286, filed Jul. 29, 1994, now abandoned.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to field of declarative, report-generation tools.

2. Background Art

In the prior art, a report writer generates textual reports or graphical reports or both from selected data in a database. The reports are produced using a report definition that includes a layout definition and a data definition. In the prior art, the report definition is accomplished by writing procedural code using programming languages or description languages, i.e., a programmer writes instructions to define a report.

Report generation tools emphasizing report layout have evolved from the script- and language-driven tools, described above. Such prior art systems, which do not require knowledge of a script or programming language, allow the definition of report layout in a declarative manner. These systems provide a graphical tool for specifying the layout of a report, i.e., the formatting of the report. To define the report layout, an input device (e.g., mouse) allows a user to position objects on a screen. The prior art declarative systems use non-programming tools for defining the overall layout of the report, i.e., the positioning of data, headers, footers, etc. Thus, the prior art focuses on the layout aspect of a report, providing visual and interactive tools so that users may create and define the properties of layout objects.

Prior art systems have not focused on the data aspect of a report, and in fact, most prior art systems assume that the data for a report is obtained using a single query. A query is a Structured Query Language (SQL) statement (i.e., a "select" statement) that defines which rows and columns from specified relations or views are to be fetched from a database(s).

The following are patents related to tools for generating reports:

Litteken discloses a method of formatting graphics in a document using a textual markup language and a tag. A tag associates a series of formats for the associated portion of the document. The particular tag described in Litteken provides for the drawing of rectangular boxes with rounded corners to simulate computer screens. The text enclosed within the box determines the scope of the box. A database is used to store document format information such as headings and figures.

Barker '210 and '211 disclose a facility for integrating different data types, or data objects, on a page of a document. Examples of objects include text and non-text objects such as: graphic objects, block objects (e.g., a block of characters), and table objects (e.g., characters in tabular form). Each object set has a data structure for its type. A superblock, which is a displayable area containing two or more object sets positioned so that the object sets overlap one another, allows a user to manipulate object sets as a group within a single displayable area on a page. Barker discloses an editor for manipulating a superblock or objects sets within a superblock. Examples of such manipulations include: create, move, copy, paste, get, delete, and describe.

Dunn discloses a schematic diagram generating system using a library of graphic icons with associated graphic and/or function primitives. Whenever an icon is selected in the process of building a schematic diagram, the rule set is accessed to control the employment and manipulation of the icon in the current diagram.

Barker '029 describes a method for editing documents containing mixed data objects (e.g., text, graphics, business graphics, and tables objects). The invention provides functionality for document formatting, object positioning, simple text editing, move, copy, delete, search/replace, go to, get, describe, create. Some commands may be used to perform operations on any type of object, while other commands are limited to certain object types.

Terasawa describes an apparatus for processing composite (i.e., integrated) documents. The apparatus includes multiple memories for storing document, attribute (e.g., composite document location, and the area, size, and type of data stored in document memory), window/viewport (i.e., display attributes such as size of window and viewport), and video (i.e., display or image) information. Two viewports are provided, where the first contains the entire document and the second contains some subset of the entire document. Updates made in one viewport result in updates to the image in memory, and are reflected in both viewports.

Yamaguchi refers to a system for processing composite documents (i.e., documents containing multiple types of data such as text and image). The system contains three storage sections that are used to generate a composite document: format-data, reference-data, and layout. The format-data section stores form-data such as region-attribute data. A text portion of a document can reference an image. The reference-data section stores information about the relationship between an image reference contained in text data and the referring text. The data layout contains information for laying out the text data and the image data.

Torres refers to a system for performing computations on data in a table using graphic icons that represent certain operations (e.g., sum, mean, and standard deviation) that can be performed on the data. The result of the operation can be displayed on the screen by placing the selected icon at a location on the screen.

Ferriter describes a system and method for generating manufacturing process plans. The system contains a library of icons that represent manufacturing process steps. Each icon has an associated set of stored textural instructions. Icons can be selected to detail the manufacturing process for a selected product. A visual image of the product to be manufactured is marked with reference numbers that correspond to each textual instruction.

Thus, while report-generation tools have evolved from script-and language-driven tools to visual, interactive, declarative tools, the prior art emphasizes the layout aspect of reports but not the data aspect of reports. The prior art tools allow a user to visually and interactively create and define the properties of layout objects, thereby greatly enhancing the ease with which a report is laid out. However, the prior art insufficiently addresses the data aspect of reports, with most tools disadvantageously assuming that the data for a report comes from a single query.

SUMMARY OF THE INVENTION

The present invention is a method and apparatus for declarative report definition and generation using an enhanced, graphical data model and a graphical user interface (GUI) for graphically specifying data items and data